CLAIMS

1. A method for configuring multiple processors into a multiprocessor system for operating a mobile vehicle, comprising:

operating device managers in the multiple processors that each monitor for signals from a new device not currently coupled to the multiprocessor system;

determining whether a communication protocol used by the new device is compatible with any one of multiple different protocols used in the multiprocessor system;

adding new device to the multiprocessor system when the communication protocol used by the new device is compatible with one of the multiple different protocols used in the multiprocessor system; and

using the new device to run different applications in the mobile vehicle.

2. A method according to claim 1 including adding the new device to the multiprocessor system only when a device Id for the new device is identifiable by one of the data managers.

3. A method according to claim 2 including adding applications running in the new device to a processor array in the multiprocessor system.

4. A method according to claim 1 including displaying applications running on the new device on a graphical user interface in the multiprocessor system.

5. A method according to claim 1 wherein the communication protocols used in the multiprocessor system include all of the following:

an IEEE 802.11 protocol;

a Blue tooth protocol;

an Ethernet protocol;

an Universal Serial Bus protocol;

a TCP/IP protocol;

a low earth and high earth satellite communications protocol; and

an infrared IRD protocol.

6. A method according to claim 1 including:

detecting a wireless signal output from a audio player brought into the mobile vehicle;

detecting the wireless signal by one of the multiple processors in the multiprocessor system; and

incorporating the audio player on the multiprocessor system when the wireless signal from the audio player is compatible with one of the multiple protocols used in the multiprocessor system.

7. A method according to claim 6 including:

displaying an icon on a graphical user interface that represents the audio player;

displaying on the graphical user interface any devices in the multiprocessor system that can output audio data from the audio player; and

connecting an audio output from the audio player to any of the devices selected from the graphical user interface.

8. A method according to claim 1 including:

detecting with one of the data managers a wireless signal output from a drive-through business establishment;

coupling the drive-through business establishment to the multiprocessor system with one of the communication protocols used by the multiprocessor system;

displaying an icon from a graphical user interface in the mobile vehicle representing the drive-through business establishment;

displaying a menu of selections from the drive-through business establishment when the icon is selected; and

transferring selections from the menu between the multiprocessor system and the drive-through establishment seamlessly, regardless of the physical data link.

9. A method according to claim 1 including:

monitoring applications running on the multiple processors in the multiprocessor system;

detecting when one of the applications fails;

automatically moving a copy of the failed application to another one of the multiple processors in the multiprocessor system;

running the copy from the other one of the multiple processors; and

performing reconfiguration to spare modules to enable for N+1 sparing in the processing system.

10. A method according to claim 9 including:

identifying critical and non-critical applications running on the processors; and

replacing one of the identified non-critical applications with a copy of the failed application.

11. A multiprocessing system for an automobile, comprising:

multiple processors that run different automobile applications;

multiple links that couple the multiple processors together; and

a dynamic configuration system operating in at least some of the multiple processors that automatically incorporates new devices into the multiprocessing system and automatically reconfigures the multiprocessor system in real-time to run the automobile applications on different processors in the multiprocessing system.

12. A multiprocessing system according to claim 11 wherein the dynamic configuration system includes a device manager that detects signals generated by new devices and incorporates the new devices into the multiprocessor system when the signals conform a protocol used between the multiple processors.

13. A multiprocessing system according to claim 11 wherein the dynamic configuration system includes a configuration manager that tracks the different applications operating in the different processors and automatically reconfigures the multiprocessing system to run failed applications on different ones of the multiple processors.

14. A multiprocessing system according to claim 13 including storing a copy of the application that has failed and downloading and running the stored copy of the application to another processor when the failure is detected.

15. A multiprocessing system according to claim 14 including storing critical data generated by the failed application and downloading and running the stored critical data along with the copy of the application on another processor.

16. A multiprocessing system according to claim 13 including displaying applications that have failed and then displaying applications in the other processors that can be replaced with copies of the failed applications.

17. An automobile processing system according to claim 16 including identifying types of data transferred by different devices in the multiprocessing system and displaying the different devices in the multiprocessing system that can output the identified types of data.

18. An automobile processing system according to claim 16 including performing the following applications with the multiprocessor system:

automatic brake control;

audio player control;

video player control;

airbag deployment monitoring;

display control;

navigation control; and

sensor monitoring.

19. A method for configuring real-time applications in a distributed multiprocessor system, comprising:

identifying applications running in multiple processors in the multiprocessor system;

detecting a failure in one of the applications in the multiprocessor system;

automatically identifying another processor in the multiprocessor system for running the failed application;

moving the failed application to the other processor; and

initiating the application in the other processor.

20. A method according to claim 19 including replacing one or more of the applications in the multiprocessing system with the failed application when there is no extra processing capacity in the multiprocessing system for running the failed application in another processor.

21. A method according to claim 20 including identifying critical and non-critical applications running in the multiprocessor system and replacing one of the identified non-critical applications with a copy of the failed application.

22. A method according to claim 19 including:

identifying a new device that is not currently coupled to the multiprocessor system;

coupling the new device to the multiprocessor system when output signals from the new device conform to a communication protocol used in the multiprocessor system;

detecting new applications running on the new device; and

adding the new applications running on the new device to a stored list of all applications running in the multiprocessor system.

23. A method according to claim 22 including:

identifying a data type used by the new device;

identifying output devices in the multiprocessor system that uses the same data type;

displaying on a graphical interface the new device and the identified output devices; and outputting data from the new device to the output devices selected on the graphical interface.

24. A method according to claim 19 including:

identifying a failed application to a car operator;

identifying non-critical applications that can be replaced by the failed application;

detecting one of the non-critical applications selected by a car operator;

replacing the selected non-critical application with a copy of the failed application; and

running the copy of the failed application.

25. A method according to claim 19 including:

storing a critical application in memory;

storing data generated by the critical application in memory; and

downloading the stored critical application and the stored critical data to another processor when the application fails.

26. A method according to claim 19 including:

moving a portable transmitting device within communication range of the multiprocessor system;

detecting a protocol used by the portable device;

automatically configuring the portable device into the multiprocessor system when the protocol conforms with one of the protocols used in the multiprocessor system;

displaying an application running on the portable device from an in-dash display coupled to the multiprocessor system; and operating the application on the portable device from the in-dash display.

27. A method according to claim 26 including using different wireless and hardwired communication protocols to communicate between different processors in the multiprocessor system.

28. A method according to claim 27 including controlling car braking, car airbag deployment, car sensor monitoring, and car audio processing with the multiprocessing system.

29. A multiprocessor system used in a car, comprising:

multiple processors adapted to run different real-time car applications;

different communication links coupling the multiple processors together; and

a dynamic configuration system run on the multiple processors that includes a device manager for automatically detecting and adding new devices to the multiprocessor system, a configuration manager that automatically reconfigures the multiprocessor system to run the real-time car applications on different ones of the multiple processors, and a data manager that identifies data generated by the new devices and identifies other devices in the multiprocessor system that can input or output the identified data.

30. A multiprocessor system according to claim 29 wherein the real-time car applications include any of the following:

car braking;

audio control;

video control;

car sensor monitoring;

car display control;

car security monitoring;

car temperature control;

car lighting control; and

car airbag monitoring.

31. A multiprocessor system according to claim 29 wherein the different communication links include a IEEE 802.11 link, a blue tooth link, and a packet based hardwired link, a satcom link, and a cellular link.

32. A multiprocessor system according to claim 29 including memory for storing:

a list of the applications running in the multiprocessor;

backup copies of selected applications running on the multiple processors; and

data generated by some of the applications.